## REMARKS

Claims 1-33 are pending. As noted in the Office Action, the applicants have elected to prosecute claims 1-9 and 25-33 in response to the previous restriction requirement. Accordingly, the applicants have canceled claims 10-24 but reserve the right to pursue those claims in a divisional application.

Paragraphs 10 and 19 of the specification have been amended to update the status of the co-pending application referenced therein.

Claims 1, 2, 9, 25, 26 and 33 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent 6,671,687 (hereinafter "Pederson") in view of U.S. Patent No. 6,785,690 (hereinafter "Davidson"). Claims 3-8 and 27-32 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Pederson and Davidson and further in view of U.S. Patent No. 6,070,174 (hereinafter "Starek"). The applicants respectfully submit that claims 1-9 and 25-33 patentably define over Pederson, Davidson and Starek, alone or in combination, for the following reasons.

### *The Present Invention*

The present invention generally relates to storing and retrieving an instance (*i.e.* an object) of a user defined type to and from a database store in which the user defined type has been defined. In the context of a database system, a user defined type (UDT) is a mechanism by which a user can create a new data type that is more complex than the native, scalar data types supported by the database system. A user defined type typically comprises a number of individual fields, each defined as one of the native data types of the database system (*e.g.,* string, integer, etc.). Figure 6 of the present application shows the definition of a user defined type called "Person" that has two fields, "FirstName" and "PhotoFS." The "FirstName" field is defined as the native data type SqlString, and the "PhotoFS" field is defined as the native data type SqlBytes. The "FirstName" field of an instance of this Person UDT may hold the characters of the name of a person, and the "PhotoFS" field may hold the data of a digital image of the person.

According to the present invention, when a user defined type is created or defined within a database system, one or more fields of the user defined type definition may be designated to be stored as a file outside of the database store and separately from the other

fields of the user defined type. Then, when a request is received to store an instance (*i.e.*, an

object) of the user defined type in the database store, the data in any so designated field of the

object will instead be stored as a file outside of the database store, preferably within a file

system of the computer on which the database store is implemented. The data in each of the

other fields of the object is stored within the database store in the usual manner. The

database store maintains a link, or reference, between the persisted object in the database

store and the data of the field that is stored as a file outside of the database store. For

example, in the case of an instance of the "Person" UDT shown in Figure 6, the data in the

"FirstName" field of the instance may be stored within the database, but the digital image

data of the "PhotoFS" field may be stored in a separate file outside of the database store. In

place of the image data, the database store will instead store a link, or reference, to the file

that contains that data outside of the database store.

The concept of storing the data of some fields of an instance of a user defined type

within a database store while storing the data of other fields of the instance of the user

defined type as a separate file outside of the database store is reflected in each of the

independent claims of the present application. Specifically, claim 1 recites:

> 1.     In a system in which an object that is an instance of a
> user defined type can be persisted in a database store, wherein a
> definition of the user defined type comprises one or more fields
> and behaviors, each field having a respective data type, *at least*
> *one of said fields of the definition being designated as*
> *containing data of a type that is to be stored as a file outside*
> *of the database store separately from the other fields of the*
> *type definition*, a method comprising:
>
> receiving a request to store an object that is an instance of the
> user defined type;
>
> storing the data in *said at least one designated field* of the
> instance of the object *as a file outside of the database store*;
> and
>
> storing the data in each of the *other fields* of the instance of the
> object *within the database store*.

Independent claim 25 recites similar features. None of the references cited in the Office

Action, alone or in combination, teaches the claimed concept of storing the data of some

fields of an instance of a user defined type within a database store while storing the data of
one or more other fields of the instance as a separate file outside of the database store.

### *Neither Pederson, Davidson nor Starek Teaches or Suggests the Claimed Invention*

The Office Action asserts that Pederson teaches the concept of storing the data of a
designated field of an instance of a user defined type outside of a database store while storing
the data of the other fields within the database store. That is not correct. Pederson teaches
the notion of assigning a user defined data type to certain data in a database system in order
to associate an access restriction with the data so that access to the data by other programs on
a network can be restricted. *See*, col. 2, ll. 5-25; col. 3, ll. 15-18. None of the portions of the
Pederson reference cited in the Office Action describes the applicants' claimed concept of
storing the data of *some fields* of an instance of a user defined type *within a database store*
while storing the data of one or more *other fields* of the instance *as a separate file outside of
the database store*.

Davidson discloses that "a class specification, describing data items of a particular
form desired to be stored, is designed and placed in [an] object catalog," and that "[a]n
instance object of the class type is created for each data item desired to be stored, and put in
the persistent store by the storage manager." Col. 2, ll. 25-32. The Office Action relies upon
Davidson only for the assertion that it teaches "receiving a request to store an object that is an
instance of [a] user defined type." While this assertion may be correct, Davidson does not
teach or suggest the applicants' claimed concept of storing the data of some fields of an
instance of a user defined type within a database store while storing the data of one or more
other fields of the instance as a separate file outside of the database store.

Starek also does not appear to disclose the concept of storing the data of a designated
field of an instance of a user defined type outside of a database while storing the data of the
other fields within the database store.

Because none of the cited references, alone or in combination, teach or suggest the
concept recited in independent claims 1 and 25 of storing the data of *some fields* of an
instance of a user defined type *within a database store* while storing the data of one or more
*other fields* of the instance *as a separate file outside of the database store*, the applicants
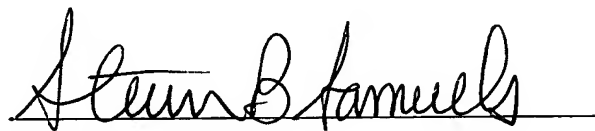respectfully submit that claims 1 and 25 patentably define over the cited references.

Inasmuch as the remaining claims depend either directly or indirectly from one of claims 1 and 25, the applicants respectfully submit that they too patentably define over the cited art of record for the same reasons. Reconsideration of the Section 103(a) rejections of claims 1-9 and 25-33 and an early Notice of Allowance are respectfully requested.

## CONCLUSION

For all the foregoing reasons, the applicants respectfully submit that the present application is in condition for allowance. Reconsideration of the Office Action and an early notice of allowance are respectfully requested.

Respectfully submitted,

Date: April 1, 2005

Steven B. Samuels
Registration No. 37,711

Woodcock Washburn LLP
One Liberty Place - 46th Floor
Philadelphia PA 19103
Telephone: (215) 568-3100
Facsimile: (215) 568-3439